## Claims

**What is claimed is:**

1. A method for compressing multi-dimensional data comprising the steps of:

a) receiving the multi-dimensional data, the multi-dimensional data comprising a plurality of data vectors indicative of an image of an object;

b) partitioning the plurality of data vectors into at least two clusters based on similarity of the data vectors such that each of the at least two clusters contains similar data vectors; and,

c) providing each of the at least two clusters to at least a compression engine for processing.

2. A method for compressing multi-dimensional data as defined in claim 1 wherein the data vectors are partitioned in a geometrically irregular fashion.

3. A method for compressing multi-dimensional data as defined in claim 1 wherein the at least a compression engine comprises at least two compression engines and comprising the step c1) of assigning each of the at least two clusters to a respective compression engine of the at least two compression engines for simultaneously processing the at least two clusters.

4. A method for compressing multi-dimensional data as defined in claim 1 comprising the step of processing the at least two clusters by performing the steps of:

d) determining a plurality of codevectors through training for approximating each of the data vectors of a cluster of the at least two clusters with a fidelity above a predetermined threshold based on the data vectors contained in the cluster; and,

e) encoding each of the data vectors of a cluster using a codevector of the plurality of codevectors.

5. A method for compressing multi-dimensional data as defined in claim 4 comprising the steps of:

f) storing the plurality of codevectors in a codebook of a cluster; and,

g) storing in an index map of a cluster an index for each of the data vectors in the cluster indicative of a codevector's location within the codebook of the cluster.

6. A method for compressing multi-dimensional data as defined in claim 5 wherein the steps d) to g) are implemented using SAMVQ.

7. A method for compressing multi-dimensional data as defined in claim 5 comprising the step of providing the index map and the codebook for transmission.

8. A method for compressing multi-dimensional data as defined in claim 1 wherein the size of the at least two clusters is approximately similar within predetermined limits of difference.

9. A method for compressing multi-dimensional data as defined in claim 8 comprising the step of:
b1) adaptively controlling the size of each of the at least two clusters by splitting and merging the at least two clusters.

10. A method for compressing multi-dimensional data as defined in claim 1 wherein the data vectors are partitioned into the at least two clusters based on the distance of a data vector to the centroid of each of the at least two clusters.

11. A method for compressing multi-dimensional data as defined in claim 4 wherein the multi-dimensional data comprises data vectors of a regional data subset of a continuous data flow, the data vectors being indicative of a region of the image of an object.

12. A method for compressing a continuous data flow comprising the steps of:
a) receiving a $m^{th}$ regional data set of the continuous data flow, the $m^{th}$ regional data set comprising a plurality of data vectors indicative of a $m^{th}$ region of an image of an object;
b) generating a $m^{th}$ regional training set for codevector training comprising the plurality of data vectors of the $m^{th}$ regional data set and data vectors of a data subset adjacent to the $m^{th}$ regional data set of the $m\text{-}1^{th}$ regional data set;

c) determining a plurality of codevectors through training for approximating each of the data vectors of the $m^{th}$ regional data set with a fidelity above a predetermined threshold based on the data vectors contained in the $m^{th}$ regional training set;

d) encoding each of the data vectors of the $m^{th}$ regional data set using a codevector of the plurality of trained codevectors;

e) providing the plurality of trained codevectors in a $m^{th}$ regional codebook; and,

f) providing in a $m^{th}$ regional index map an index indicative of a codevector's location within the $m^{th}$ regional codebook.

13. A method for compressing a continuous data flow as defined in claim 12 comprising the steps of:

e1) storing the plurality of trained codevectors in a $m^{th}$ regional codebook; and,

f1) storing in a $m^{th}$ regional index map an index for each of the data vectors of the $m^{th}$ regional data set indicative of a codevector's location within the $m^{th}$ regional codebook.

14. A method for compressing a continuous data flow as defined in claim 13 comprising the step of:

g) providing the $m^{th}$ regional index map and the $m^{th}$ regional codebook for transmission.

15. A method for compressing a continuous data flow as defined in claim 14 comprising the step of repeating steps a) to g) for subsequent regions.

16. A method for compressing a continuous data flow as defined in claim 15 wherein the data subset of the $m\text{-}1^{th}$ regional data set comprises data vectors corresponding to at least an image line.

17. A method for compressing a continuous data flow as defined in claim 1 comprising the steps of:

b1) partitioning the plurality of data vectors of the $m^{th}$ regional training set into at least two clusters based on similarity of the data vectors such that each of the at least two clusters contains similar data vectors; and,

b2) providing each of the at least two clusters to at least a compression engine for processing.

18. A method for compressing a continuous data flow as defined in claim 12 wherein the data subset equals the $m$-$1^{th}$ regional data set.

19. A method for compressing a continuous data flow as defined in claim 18 wherein the plurality of codevectors is determined for approximating each of the data vectors of the $m^{th}$ regional data set and the $m$-$1^{th}$ regional data set, and wherein each of the data vectors of the $m^{th}$ regional data set and the $m$-$1^{th}$ regional data set are encoded.

20. A method for compressing a continuous data flow as defined in claim 18 comprising the steps of:

e) storing the plurality of trained codevectors in a $m^{th}$ regional codebook;

f) storing in a $m^{th}$ regional baseline index map an index indicative of a codevector's location within the $m^{th}$ regional codebook, the index corresponding to a data vector of the $m^{th}$ regional data set; and,

g) storing in a $m$-$1^{th}$ regional redundant index map an index indicative of a codevector's location within the $m^{th}$ regional codebook, the index corresponding to a data vector of the $m$-$1^{th}$ regional data set.

21. A method for compressing a continuous data flow as defined in claim 20 comprising the step of:

h) providing the $m^{th}$ regional codebook, the $m^{th}$ regional baseline index map and the $m$-$1^{th}$ regional redundant index map for transmission.

22. A method for compressing a continuous data flow as defined in claim 21 comprising the step of repeating steps a) to h) for subsequent regions.

23. A method for compressing a continuous data flow in real-time as defined in claim 21 comprising the step of compressing the $m^{th}$ regional codebook, the $m^{th}$ regional baseline index map and the $m$-$1^{th}$ regional redundant index map prior transmission using lossless compression.

24. A method for compressing a continuous data flow as defined in claim 20 comprising the steps of:

b1) partitioning the plurality of data vectors of the $m^{th}$ regional training set into at least two clusters based on similarity of the data vectors such that each of the at least two clusters contains similar data vectors; and,

b2) providing each of the at least two clusters to at least a compression engine for processing.

25. A method for compressing a continuous data flow as defined in claim 22 comprising the steps of:

receiving a first regional data set of the continuous data flow, the first regional data set comprising a plurality of data vectors indicative of a first region of the image of an object;

generating a first regional training set for codevector training comprising the plurality of data vectors of the first regional data set;

determining a plurality of codevectors through training for approximating each of the data vectors of the fist regional data set with a fidelity above a predetermined threshold based on the data vectors contained in the first regional training set;

encoding each of the data vectors of the first regional data set using a codevector of the plurality of trained codevectors;

storing the plurality of trained codevectors in a first regional codebook;

storing in a first regional baseline index map an index for each of the data vectors of the first regional data set indicative of a codevector's location within the first regional codebook; and,

providing the first regional codebook and the first regional baseline index map for transmission.

26. A method for compressing a continuous data flow as defined in claim 22 comprising the steps of:

receiving the $m^{th}$ regional codebook, the $m^{th}$ regional baseline index map and the $m$-$1^{th}$ regional redundant index map via a communication link; and,

reconstructing the $m^{th}$ regional data set using the $m^{th}$ regional codebook and the $m^{th}$ regional baseline index map.

27. A method for compressing a continuous data flow as defined in claim 22 comprising the steps of:

receiving the $m^{th}$ regional codebook, the $m^{th}$ regional baseline index map and the $m\text{-}1^{th}$ regional redundant index map via a communication link; and,

reconstructing the $m\text{-}1^{th}$ regional data set using the $m^{th}$ regional codebook and the $m\text{-}1^{th}$ regional redundant index map.

28. A method for compressing a continuous data flow as defined in claim 22 wherein the steps c) to f) are implemented using SAMVQ.

29. A method for compressing a continuous data flow as defined in claim 22 wherein the data vectors comprise hyper-spectral vectors.

30. A storage medium having stored thereon at least an executable command for when executed resulting in performance of the steps of:

a) receiving the multi-dimensional data, the multi-dimensional data comprising a plurality of data vectors indicative of an image of an object;

b) partitioning the plurality of data vectors into at least two clusters based on similarity of the data vectors such that each of the at least two clusters contains similar data vectors;

c) providing each of the at least two clusters to at least a compression engine for processing;

d) determining a plurality of codevectors through training for approximating each of the data vectors of a cluster of the at least two clusters with a fidelity above a predetermined threshold based on the data vectors contained in the cluster;

e) encoding each of the data vectors of a cluster using a codevector of the plurality of trained codevectors;

f) storing the plurality of trained codevectors in a codebook; and,

g) storing in an index map an index for each of the data vectors of a cluster indicative of a codevector's location within the codebook.

31. A system for compressing multi-dimensional data comprising:

a first port for receiving the multi-dimensional data;

electronic circuitry in data communication with the first port, the electronic circuitry for performing the steps of:

a) receiving the multidimensional data, the multi-dimensional data comprising a plurality of data vectors indicative of an image of an object;

b) partitioning the plurality of data vectors into at least two clusters based on similarity of the data vectors such that each of the at least two clusters contains similar data vectors;

c) providing each of the at least two clusters to at least a compression engine for processing;

d) determining a plurality of codevectors through training for approximating each of the data vectors of a cluster of the at least two clusters with a fidelity above a predetermined threshold based on the data vectors contained in the cluster;

e) encoding each of the data vectors of a cluster using a codevector of the plurality of trained codevectors;

f) storing the plurality of trained codevectors in a codebook; and,

g) storing in an index map an index for each of the data vectors of a cluster indicative of a codevector's location within the codebook;

and,

a second port in data communication with the electronic circuitry for providing the regional codebook and the regional index map.


32. A system for compressing a continuous data flow as defined in claim 31 wherein the electronic circuitry comprises at least a processor.


33. A storage medium having stored thereon at least an executable command for when executed resulting in performance of the steps of:

a) receiving a $m^{th}$ regional data set of the continuous data flow, the $m^{th}$ regional data set comprising a plurality of data vectors indicative of a $m^{th}$ region of an image of an object;

b) generating a $m^{th}$ regional training set for codevector training comprising the plurality of data vectors of the $m^{th}$ regional data set and the plurality of data vectors of the $m\text{-}1^{th}$ regional data set;

c) determining a plurality of codevectors through training for approximating each of the data vectors of the $m^{th}$ regional data set and the $m\text{-}1^{th}$ regional data set with a fidelity above a predetermined threshold based on the data vectors contained in the $m^{th}$ regional training set;

d) encoding each of the data vectors of the $m^{th}$ regional data set and the $m\text{-}1^{th}$ regional data set using a codevector of the plurality of codevectors;

e) storing the plurality of trained codevectors in a $m^{th}$ regional codebook;

f) storing in a $m^{th}$ regional baseline index map an index indicative of a codevector's location within the $m^{th}$ regional codebook, the index corresponding to a data vector of the $m^{th}$ regional data set;

g) storing in a $m\text{-}1^{th}$ regional redundant index map an index indicative of a codevector's location within the $m^{th}$ regional codebook, the index corresponding to a data vector of the $m\text{-}1^{th}$ regional data set;

h) providing the $m^{th}$ regional codebook, the $m^{th}$ regional baseline index map and the $m\text{-}1^{th}$ regional redundant index map for transmission; and,

repeating steps a) to h) for subsequent regions.


34. A system for compressing multi-dimensional data comprising:

a first port for receiving the multi-dimensional data;

electronic circuitry in data communication with the first port, the electronic circuitry for performing the steps of:

a) receiving a $m^{th}$ regional data set of the continuous data flow, the $m^{th}$ regional data set comprising a plurality of data vectors indicative of a $m^{th}$ region of an image of an object;

b) generating a $m^{th}$ regional training set for codevector training comprising the plurality of data vectors of the $m^{th}$ regional data set and the plurality of data vectors of the $m\text{-}1^{th}$ regional data set;

c) determining a plurality of codevectors through training for approximating each of the data vectors of the $m^{th}$ regional data set and the $m\text{-}1^{th}$ regional data set with a fidelity above a predetermined threshold based on the data vectors contained in the $m^{th}$ regional training set;

d) encoding each of the data vectors of the $m^{th}$ regional data set and the $m\text{-}1^{th}$ regional data set using a codevector of the plurality of codevectors;

e) storing the plurality of trained codevectors in a $m^{th}$ regional codebook;

f) storing in a $m^{th}$ regional baseline index map an index indicative of a codevector's location within the $m^{th}$ regional codebook, the index corresponding to a data vector of the $m^{th}$ regional data set;

g) storing in a $m\text{-}1^{th}$ regional redundant index map an index indicative of a codevector's location within the $m^{th}$ regional codebook, the index corresponding to a data vector of the $m\text{-}1^{th}$ regional data set;

h) providing the $m^{th}$ regional codebook, the $m^{th}$ regional baseline index map and the $m\text{-}1^{th}$ regional redundant index map for transmission; and,

repeating steps a) to h) for subsequent regions;

and,

a second port in data communication with the electronic circuitry for providing the $m^{th}$ regional codebook, the $m^{th}$ regional baseline index map and the $m\text{-}1^{th}$ regional redundant index map.